IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

APPEAL NO. _____

First named inventor: Martin G. Rammel | Docket No. 03-0945

Serial No.10/720,614 | Filed: Nov. 24, 2003

Examiner: Thuy Chan Dao | Art Unit: 2192

Title: METHODS AND APPARATUS FOR SIMULATION SPEEDUP | Confirm. No. 4243

APPEAL BRIEF

Hugh P. Gortler, Esq.

(949) 454-0898

TABLE OF CONTENTS

1. REAL PARTY IN INTEREST

The real party in interest is the assignee, The Boeing Company.

2. RELATED APPEALS AND INTERFERENCES

No appeals or interferences are known to have a bearing on the Board's decision in the pending appeal.

3. STATUS OF CLAIMS

Claims 1-16 and 18-30 are cancelled.

Claims 17 and 31-37 are pending.

Claims 17 and 31-37 are rejected.

The rejections of claims 31-37 are being appealed.

4. STATUS OF AMENDMENTS

No amendment was filed subsequent to the final office action dated 13 January 2010.

5. SUMMARY OF CLAIMED SUBJECT MATTER

Consider a numerical simulation of a radar receiver processor. At the time the application was filed, the numerical simulation could take approximately two weeks of CPU time on a modern high-speed computer to provide 1.6 seconds of real-time radar data (page 2, lines 25-30 of the application).

Base claims 31 and 35 recite a method and system for improving the speed of a numerical simulation. The speed is improved by offloading a portion of the simulation from a computer to an FPGA, and managing the dataflow between the two. Thus, instead of the computer running the entire simulation, a first portion is performed by the computer, and a second portion is performed by the FPGA.

Base claim 31

Base claim 31 recites a method of performing a numerical simulation with a Field Programmable Gate Array (FPGA) and a separate central processing unit (CPU). Page 3, lines 10-16 refer to the FPGA as a programmable device and the CPU as a serial processor. Page 3, line 20 uses the term "serial CPU."

An embodiment of the method 200 is illustrated in Figure 2 (page 4, lines 23-24). The method 200 includes using the CPU to perform a numerical simulation including generating input signals 212 and 214 (page 3, lines 12-13) and sending the input signals 212 and 214 to the FPGA (page 3, lines 13-14; and page 5, lines 16-17). The method further includes using the FPGA to apply a model 226 to the input signals 212 and 214 and send results of the model 226 back to the CPU (page 3, lines 14-16; page 4, lines 17-22; and page 5, lines 16-25).

The FPGA also generates a first output (vout) that marks data as valid or invalid (page 5, lines 27-28), a second output (done) that indicates the first sample

of each frame (page 5, line 29 to page 6, line 1), and a third output (rfd) that indicates when the model can accept data (page 6, lines 1-2).

The CPU uses the results in the numerical simulation and the outputs to maintain data flow with the FPGA (page 4, lines 17-22).

Base claim 35

Base claim 35 recites apparatus comprising a central processing unit (CPU) and a Field Programmable Gate Array (FPGA) for performing different portions of a numerical simulation. For example, the apparatus is a computer, which includes a CPU (page 4, lines 21-22) and a PC card (page 4, line 14). In one embodiment, the PC card is a PCI bus card 218, and the FPGA is included in the PCI bus card 218 (page 5, lines 7-9).

The CPU is programmed to perform a numerical simulation of sine wave functions representing real and imaginary inputs 212 and 214 (page 3, lines 12-13). The FPGA is programmed to perform a Fast Fourier Transform (FFT) on the inputs 212 and 214 and send results of the FFT back to the CPU (page 3, lines 14-16; page 4, lines 17-22; and page 5, lines 16-25).

The FPGA also generates a first output (vout) that marks data as valid or invalid (page 5, lines 27-28), a second output (done) that indicates the first sample of each frame (page 5, line 29 to page 6, line 1), and a third output (rfd) that indicates when the model can accept data (page 6, lines 1-2).

The CPU uses the results in the numerical simulation and the outputs to maintain data flow with the FPGA (page 4, lines 17-22).

Dependent claims 34 and 37

Claim 34 depends from base claim 31, and claim 37 depends from base claim 35. Claims 34 and 37 both recite that the simulation is a radar simulation. Functions for the radar simulation are described in Figure 1 and page 2, lines 4-23. In the claimed method and system, some of the elements are run in the FPGA instead of the CPU (page 4, lines 17-20).

6. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

a. Rejection of claim 31 under 35 USC §102(e) as being anticipated by Ballagh U.S. Patent No. 6,883,147.

b. Rejection claims of 32-37 under 35 USC §103(a) as being unpatentable over Ballagh in view of admitted prior art.

7. ARGUMENTS

Ballagh's Figure 2 illustrates a circuit design 200 including an FPGA 202 having an on-chip processor 204, on-chip bus 206, on-chip interface 208, and on-chip peripheral component 210 (page 5, lines 4-9). The peripheral component 210 includes a reconfigurable digital filter, filter control logic, and logic for interfacing with the bus 206 (col. 5, lines 19-22). The peripheral component 210 can communicate with a host computer (external processor) via the interface 208 (col. 5, lines 22-23).

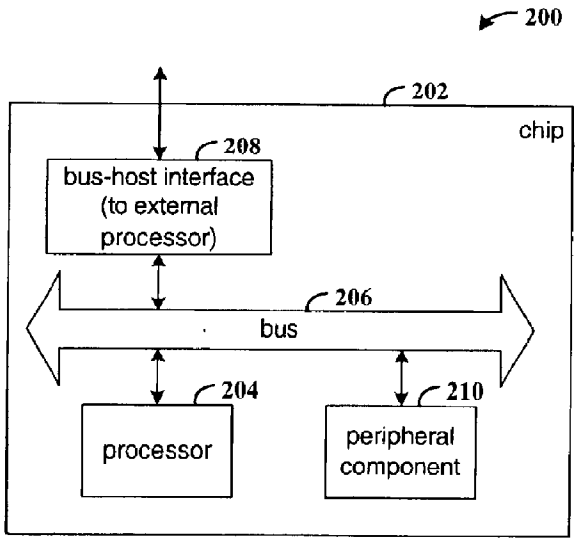

FIG. 2

The reconfigurable digital filter is described as a finite impulse response (FIR) filter (col., 3, lines 54-55; and page 5, line 39). Filter coefficients are loaded into the FIR filter, which applies the filter coefficients to input values (col. 5,lines 55-56). The filter coefficients are reloadable (col. 5, line 54).

The filter coefficients are supplied to the FPGA 202 by the host computer (col. 5, lines 24-26). The filter coefficients are received and reloaded by the on-chip processor 204 (col. 5, lines 26-28 and 36-37).

The host computer also receives filtered frames from the FPGA 202 (col. 5, lines 33-34). The on-chip processor controls data flow between the FIR filter and the host computer (col. 3, lines 56-59).

The remainder of Ballagh is devoted to describing a system and process for designing the FPGA. Figure 4a and subsequent figures illustrate a process for creating the peripheral component 210 (col. 2, lines 51-53). Figure 1 illustrates a system for designing the peripheral component (col. 2, lines 40-41 col. 3, lines 63-65).

An overview of the design process is described in Ballagh's Background section. The design process involves three phases: a specification phase, a modeling phase, and an implementation phase (col. 1, lines 17-22). The specification phase involves defining functional requirements (col. 1, lines 19-20). The modeling phase involves capturing the design in executable form, simulating, and analyzing the results (col. 1, lines 31-32). The implementation phase involves creating a low-level hardware realization in terms of primitives in an appropriate technology library (col. 1, lines 47-58). Hardware languages such as VHDL are commonly used, as are libraries of intellectual property (IP) blocks.

What does Ballagh have to do with speeding up a numerical simulation? Nothing.

I

REJECTION OF CLAIM 31 UNDER 35 USC §102(E) AS BEING ANTICIPATED BY BALLAGH U.S. PATENT NO. 6,883,147

Legal error in the '102 rejection

Claim 31 is not being given its broadest reasonable interpretation consistent with the specification. See the Federal Circuit's *en banc* decision in Phillips v. AWH Corp., 415 F.3d 1303, 75 USPQ2d 1321 (Fed. Cir. 2005).

The final office action appears to interpret a numerical simulation to be a fixed or floating point operation. However, that interpretation is not consistent with the specification. According to page 1, lines 19+ of the specification, numerical simulations are used to model different types of physical phenomena. In the field of aerospace engineering, numerical simulations can be used to predict electromagnetic scattering from reflective bodies. The detailed description provides a specific example: a radar simulation. A fixed or floating point operation is but one operation in a numerical simulation.

Moreover, the interpretation of claim 31 is not consistent with the interpretation that those skilled in the art would reach, as required by MPEP 2111. Ballagh states that arithmetic values can be represented by fixed or floating point values during a numerical simulation (col. 1, lines 39-42). Ballagh does not state that a fixed or floating point operation is a numerical simulation.

The final office action provides no other evidence that a person skilled in the art would consider a fixed or floating point operation to be a numerical simulation. Page 2 of the final office action provides evidence that floating and fixed point operations are methods of writing numeric quantities, and that a binary

point is a symbol that separates the integral part of a numerical expression from its fractional part. That evidence is silent about numerical simulations.

The interpretation of "numerical simulation" is based on legal error. For this reason alone, the '102 rejection of claim 31 should be withdrawn.

Factual deficiencies in the '102 rejection

Ballagh provides no guidance whatsoever for speeding up a numerical simulation.

Ballagh does not describe a single feature of claim 31.Ballagh is silent about a portion of a simulation being offloaded from a processor to an FPGA, whereby the processor runs a first portion of the simulation and the FPGA runs a second portion of the simulation. Ballagh only describes a simulation for designing an FPGA, wherein the entire simulation is run on a single computer.

Ballagh does not describe the use of a CPU to perform a numerical simulation including generating input signals and sending the input signals to the FPGA. Ballagh only describes an external processor for generating filter coefficients, and an on-chip processor 204 for loading the coefficients into a FIR filter.

Ballagh's circuit 200 does not apply a model to the signals from the external processor. Ballagh's external processor supplies filter coefficients to the FPGA 202, but the FPGA does NOT perform filtering on the coefficients. Rather, it uses the coefficients to perform filtering on input values x(n).

Ballagh is silent about data flow between the external processor and the FPGA. It follows that Ballagh does not describe generating a first output that marks data as valid or invalid, a second output that indicates the first sample of each frame, and a third output that indicates when the model can accept data.

In the Response to Arguments on pages 4-5, the final office action asserts that a simulation is performed during the modeling phase of the FPGA (page 4). It then asserts that the yet-to-be-designed FPGA is somehow used to run a portion of the simulation. However, Ballagh is quite clear that the modeling phase is performed by the system of Figure 1. Ballagh is also quite clear that the system of Figure 1 is used to design the circuit 200 of Figure 2, including the FPGA 202. This point was raised in the previous response. The final office action still does not respond directly to this point.

According to MPEP 2131, "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). Ballagh does not describe a single feature of claim 31. Therefore, the '102 rejection of base claim 31 should be withdrawn.

II

REJECTION OF 32-37 UNDER 35 USC §103(A) AS BEING UNPATENTABLE OVER BALLAGH IN VIEW OF ADMITTED PRIOR ART

The '103 rejection does not comply with MPEP 2142 and the U.S. Supreme Court's holding in KSR International Co. v. Teleflex Inc., 82 USPQ2d 1385, 1395-97 (2007). According to the Supreme Court, "rejections on obviousness cannot be sustained with mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness."

As discussed above, Ballagh provides no guidance whatsoever for speeding up a numerical simulation. And, as discussed in Argument I, Ballagh does not teach or suggest running a first portion of a simulation in a processor and a second portion of the simulation in an FPGA. Ballagh does not use the processor portion of the simulation to generate input signals, and the FPGA portion to process the input signals and return the input signals to the processor. Ballagh's circuit 200 does not apply a model to the signals from the external processor. Ballagh is silent about data flow between the external processor and the FPGA.

Claims 32-33 and 35-36

In any event, Ballagh does not teach or suggest offloading a specific element of a numerical simulation – the FFT - from a host computer to an FPGA. Neither does the admitted prior art. The admitted prior art describes a radar simulation in which all elements, including an FFT, are performed on a single CPU (see page 2, lines 24-29 of the present application).

The final office action does not provide articulated reasoning about offloading a specific element - the FFT – from a CPU to an FPGA. It does not explain why the FFT would be chosen over other elements of a numerical simulation. It does not explain how Ballagh's FPGA would perform the FFT. The final office action does not even provide any of the exemplary rationales to support a conclusion of obviousness (see MPEP 2143). The final office action simply offers a conclusory statement: *a person skilled in the art would be motivated to perform the FFT on an FPGA because the admitted prior art describes performing a Simulink simulation on a computer*.

Moreover, the conclusory statement does not flow from the evidence. The admitted prior art doesn't suggest running the FFT on an FPGA. The admitted prior art clearly states that all elements of the entire radar simulation, including the FFT, are performed on a single CPU (see page 2, lines 24-29 of the present application).

The final office action provides little more than a bald conclusion of obviousness. Therefore, the '103 rejections of claims 32-33 and 35-36 should be withdrawn.

Claim 34 and 37

The final office action does not explain how the host computer and FIR filter of Ballagh's system would perform the various elements of a radar simulation. The final office action does not even provide any of the exemplary rationales to support a conclusion of obviousness (see MPEP 2143). The final office action offers only a conclusory statement: a person skilled in the art would be motivated to combine the teachings of the admitted prior art into Ballagh's system because a Simulink simulation can be used to convert data from a spatial domain to a frequency domain.

Moreover, the conclusory statement does not flow from the evidence. The admitted prior art states that the entire radar simulation is performed on a single CPU (see page 2, lines 24-29 of the present application). The admitted prior art does not teach or suggest running a first portion of a radar simulation on a CPU and a second portion f the simulation on a FIR filter.

The final office action provides little more than a bald conclusion of obviousness. For this additional reason, the '103 rejections of claims 34 and 37 should be withdrawn.

For the reasons above, the rejections should be reversed. The Honorable Board of Patent Appeals and Interferences is respectfully requested to reverse the rejections.

Respectfully submitted,

/Hugh Gortler #33,890/
Hugh P. Gortler
Reg. No. 33,890

Date: April 22, 2010

8. CLAIMS APPENDIX

Claims 1-16 (Cancelled)

17. (Previously presented) The method of Claim 32, wherein the FPGA Is programmed to perform steps including:

receiving the real and imaginary inputs at first and second inputs of an FFT block via a pair of gateway in blocks;

coupling an output of a double delay block to a third input of the FFT block, the third input being adapted to mark data input as valid or invalid;

coupling an output of a k=0 block to a fourth input of the FFT block, the fourth input being adapted to control a forward or a reverse transform;

coupling outputs of the FFT block to at least one D flip flop-based registers adapted to provide a signal latency; and

coupling the outputs of the registers to at least one gateway out.

Claims 18-30 (Cancelled)

31. (Previously presented) A method of performing a numerical simulation with a Field Programmable Gate Array (FPGA) and a separate central processing unit (CPU) , the method comprising:

using the CPU to perform a numerical simulation including generating input signals and sending the input signals to the FPGA;

using the FPGA to apply a model to the input signals and send results of the model back to the CPU, the FPGA also generating a first output that marks data as valid or invalid, a second output that indicates the first sample of each frame, and a third output that indicates when the model can accept data; and

wherein the CPU uses the results in the numerical simulation and the outputs to maintain data flow with the FPGA.

32. (Previously presented) The method of claim 31, wherein the input signals include sine wave functions representing real and imaginary inputs; and wherein the model includes a Fast Fourier Transform (FFT).

33. (Previously presented) The method of claim 32, wherein the FPGA converts the real and imaginary inputs from double point precision to fixed point prior to performing the transform; and wherein the FPGA converts the results of the FFT from fixed point back to double precision prior to sending the results back to the CPU.

34. (Previously presented) The method of claim 32, wherein the CPU performs a numerical simulation of a radar system.

35. (Previously presented) Apparatus comprising a central processing unit (CPU) and a Field Programmable Gate Array (FPGA) for performing different portions of a numerical simulation;

the CPU programmed to perform a numerical simulation of sine wave functions representing real and imaginary inputs;

the FPGA programmed to perform a Fast Fourier Transform (FFT) on the inputs and send results of the FFT back to the CPU, the FPGA also generating a first output that marks data as valid or invalid, a second output that indicates the first sample of each frame, and a third output that indicates when the FFT can accept data;

the CPU using the results in the numerical simulation and the outputs to maintain data flow with the FPGA.

36. (Previously presented) The apparatus of claim 35, wherein the FPGA converts the real and imaginary inputs from double point precision to fixed point prior to performing the transform; and wherein the FPGA converts the results of the FFT from fixed point back to double precision prior to sending the results back to the CPU.

37. (Previously presented) The apparatus of claim 35, wherein the CPU performs a numerical simulation of a radar system.

9. EVIDENCE APPENDIX

None

10. RELATED PROCEEDINGS APPENDIX

None